https://dx.doi.org/10.22161/ijaers.4.2.1 *ISSN: 2349-6495(P) | 2456-1908(O)*

Comparative study of AODV with malicious environment and the USOR

Dr K Butchi Raju

Professor, Department of CSE, GRIET, Hyderabad, India

Abstract*—By the performance analysis of two protocols AODV and USOR implemented in ns2 we made a comparison between them. USOR is efficient as it uses a novel combination of group signature and ID-based encryption for route discovery. Security analysis demonstrates that USOR can well protect user privacy against both inside and outside attackers. Successful implementation of unlinkability and unobservability property of USOR not only has satisfactory performance compared to AODV, but also achieves stronger privacy protection than the existing systems when in the malicious environment. Usage of the stronger encryption techniques in unobservable protocol makes the more data secure. In this paper we are going to compare the protocols AODV and USOR. The performance of the network mainly refers by using the packet delivery function and the over head of the packet to reach the destination. Here we are analyze overhead and packet delivery function of the two protocols and made the comparison between them.*

Keywords— USOR, AODV, ns2.

I. INTRODUCTION

Nowadays wireless mobile nodes are becoming more and more capable and have improved a lot over those available in the past. In Ad hoc networks all the wireless mobile devices will be capable to communicate with each other in the absence of infrastructure. Ad hoc network allows all wireless devices within range of each other without involving any central access point and administration. Routing protocols are challenging to design as performance degrades with the growth of number of nodes in the environment and a large ad hoc network is difficult to manage, and there are more number of chances to attack by the hackers. So the main problem in the MANET[1] is providing the security to the all part of the network. To avoid security problems there are so many researchers invented many security methods like encryption methods, secure routing protocols. In our project we are going to compare the two protocols AODV[2] and USOR[3]. In AODV routes are discovered as on-demand basis and are maintained as long as they are required, and it maintains a sequence number, which it increases each time it finds a change in the topology of its neighborhood. This sequence number ensures that the most recent route is selected for execution of the route discovery. AODV is able to provide uncast, multicast and broadcast communication ability. Combination of the three makes it an advantage protocol. Route tables used by AODV store the destination and next hop IP addresses as well as the destination sequence number. AODV also provide quick deletion of invalid routes in response the route ERROR messages generated due to link breakage. If a node fails to receive three consecutive HELLO messages from a neighbor, it is concluded that link is broken for the specific node and a RERR message is broadcasted to any upstream node. In fact a more conservative routing table and sequence number driven approach is utilized in AODV. AODV is best in routing procedure but in the case of security providing to the node and data transmission there are some faults occurred. AODV failed to provide the secure data transmission. For this An Unobservable Secure On-Demand Routing is introduced. This unobservable secure routing scheme offers complete unlink ability and content un-observability[4]. USOR is efficient as it uses a novel combination of group signature and ID-based encryption for route finding. To improve security here we are using popular two methods, one is RSA algorithm[5] and Sha-1 algorithm[6]. In this project we suggested un-observability by providing protection on request and reply. Security analysis demonstrates that USOR can well protect user privacy against both inside and outside attackers. In this paper we are going to compare the both the protocols AODV and USOR with the hacking environment. Comparison is done by the over head and packet delivery functions of the both protocols.

II. ROUTING PROTOCOLS

Ad-hoc On-Demand Distance Vector Routing Protocol:

Ad-hoc On-demand distance vector routing protocol (AODV)[7,8] is a reactive routing protocol. In AODV, the network is silent until a connection is needed. At that point the network node that needs a connection broadcasts a request for connection. Other AODV nodes forward this message, and record the node that they heard it from, creating an explosion of temporary routes back to the needy node. When a node receives such a message and already has a route to the desired node, it sends a message backwards through a temporary route to the requesting node. The needy node then begins using the route that has the least number of hops through other nodes. Unused entries in the routing

https://dx.doi.org/10.22161/ijaers.4.2.1 *ISSN: 2349-6495(P) | 2456-1908(O)*

tables are recycled after a time. When a link fails, a routing error is passed back to a transmitting node, and the process repeats. Nodes uses sequence number so that they do not repeat route requests that they have already passed on. The basic operation of AODV includes the two main steps-

1. **Path Discovery**
2. **Path Maintenance.**

1. Path Discovery:

The Path Discovery process is initiated whenever a source node wants to transmit data to the destination and it has no valid routing information. Here, each node maintains two separate counters. < node sequence number and broadcast id > The sequence number is to determine the freshest route in the network. Broadcast id is initiated by the source node and it is incremented when broadcast starts from the node. The source node initiates path discovery by broadcasting a route request (RREQ) packet to its neighbors. Figure 1 represents the flow of RREQ in the network from source to the destination node. The contents of RREQ packet are: <Source IP address, source sequence number, broadcast id, destination IP address, destination sequence number, hop count> The pair < source IP address, broadcast id> uniquely identifies a RREQ. Whenever a node receives multiple copies of RREQ from the different intermediate nodes, it keeps the first RREQ packet and ignores all other RREQs. The intermediate node can reply to the source node if it has a route to the destination with equal or greater sequence number than the destination sequence number in the RREQ packet.

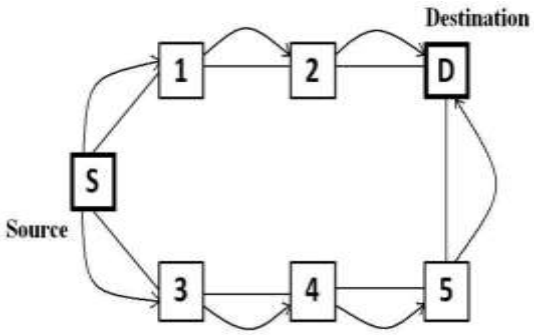

Fig.1: Route Request propagation

The routing path can be established in two steps- reverse path set up and forward path setup. The reverse path is established with the propagation of the route reply packets (RREP) in the network from the destination to the source node. When the RREQ is sent in the network, the intermediate nodes forward the RREQ after increasing the number of hops in the RREQ packet by one and also they record the address of the node from which they receive the first RREQ packet. Once the RREQ is reached at the destination node, the eligible intermediate nodes as well as the destination node propagate RREP from the destination to the source. Once the RREP reaches the source node, it establishes the reverse path. Figure 2 shows the propagation of RREP in the network from destination to the source node. The content of RREP is: <Destination IP address, source IP address, number of hops, expiration time, destination sequence number> The reverse path routing information is maintained only till the reverse path is established and this duration is represented by the expiration time. Once the reverse path is established, the forward path is established by the means of RREP propagation as the intermediate nodes record the address of the previous nodes in reverse path from destination to source node in a similar manner as the reverse path setup.

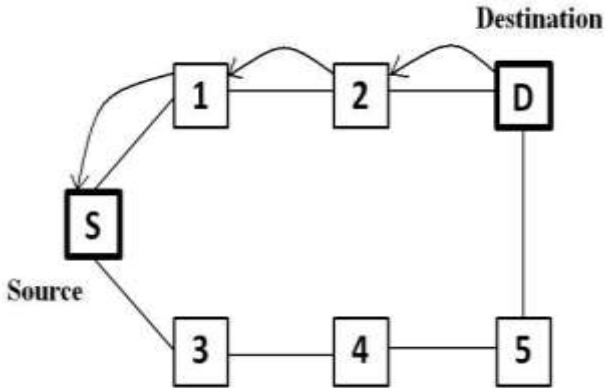

Fig.2: Route Reply

2. Path Maintenance:

The route from source node to destination is affected by the movement of active nodes lying on that path. If the source node moves during an active session, it can reinitiate the route discovery procedure. On the other hand, when the destination or some intermediate node moves, the communication link fails. So, to handle the link failure problem, the node that detects unreachable node or broken link, sets infinity as number of hops in RREP and also attach the link failure notification message (RERR) to each of its active upstream neighbor on underlying path. Once RERR reaches the source, it reinitiates the route discovery procedure. Local connectivity among the nodes can be maintained with the help of periodic broadcasting of HELLO messages but this increases traffic overhead in the network. Advantage of AODV is routes are established on demand and destination sequence numbers are used to find the latest route to the destination. Lower delay for connection setup, Disadvantage is, it doesn't allow handling unidirectional links. Multiple Route Reply packets in response to a single Route Request packet can lead to heavy control overhead. Periodic beaconing leads to unnecessary bandwidth consumption.

Unobservable Secure On-Demand Routing Protocol[9,10]:

A number of schemes have been proposed to protect privacy in ad hoc networks. However, none of these schemes offer complete unlinkability or unobservability property since data packets and control packets are still linkable and

https://dx.doi.org/10.22161/ijaers.4.2.1 *ISSN: 2349-6495(P) | 2456-1908(O)*

distinguishable in these schemes. This provides stronger privacy requirements regarding privacy-preserving routing in mobile ad hoc networks. SOR is efficient as it uses a novel combination of group signature and ID-based encryption for route discovery. Security analysis demonstrates that USOR can well protect user privacy against both inside and outside attackers.

In this protocol, both control packets and data packets look random and indistinguishable from dummy packets for outside adversaries. Only valid nodes can distinguish routing packets and data packets from dummy traffic with inexpensive symmetric decryption. The intuition behind the proposed scheme is that if a node can establish a key with each of its neighbors, then it can use such a key to encrypt the whole packet for a corresponding neighbor. The receiving neighbor can distinguish whether the encrypted packet is intended for itself by trial decryption. In order to support both broadcast and unicast, a group key and a pairwise key are needed. As a result, USOR comprises two phases: anonymous trust establishment and unobservable route discovery. The unobservable routing scheme USOR aims to offer the following privacy properties.

Anonymity: the senders, receivers, and intermediate nodes are not identifiable within the whole network, the largest anonymity set.

Unlinkability: the linkage between any two or more IOIs from the senders, the receivers, the intermediate nodes, and the messages is protected from outsiders. Note linkage between any two messages, e.g., whether they are from the same source node, is also protected.

Unobservability: any meaningful packet in the routing scheme is indistinguishable from other packets to an outside attacker. Not only are the content of the packet but also the packet header like packet type protected from eavesdroppers. And any node involved in route discovery or packet forwarding, including the source node, destination node, and any intermediate node, is not aware of the identity of other involved nodes (also including the source node, the destination node, or any other intermediate nodes).

The unobservable routing scheme comprises of two phases: anonymous key establishment as the first phase and the route discovery process as the second phase.

1) Anonymous Key Establishment: In this phase, every node in the ad hoc network communicates with its direct neighbors within its radio range for anonymous key establishment.

2) Privacy-Preserving Route Discovery: This phase is a privacy-preserving route discovery process based on the keys established in previous phase. Similar to normal route discovery process, our discovery process also comprises of route request and route reply. The route request messages flood throughout the whole network, while the route reply messages are sent backward to the source node only.

Suppose there is a node *S* (source) intending to find a route to a node *D* (destination), and *S* knows the identity of the destination node *D*. Without loss of generality, we assume three intermediate nodes between *S* and *D*, as illustrated in Fig. 3. The route discovery process executes as follows:

Route Request (RREQ): *S* chooses a random number *rS*, and uses the identity of node *D* to encrypt a trapdoor information that only can be opened with *D*'s private Idbased key, which yields *ED*(*S,D, rSP*). *S* then selects a sequence number *seqno* for this route request, and another random number *NS* as the route pseudonym, which is used as the index to a specific route entry. To achieve unobservability, *S* chooses a nonce *NonceS* and calculates a pseudonym as Nym*S*=*H*3(*kS*∗|*NonceS*). Each node also maintains a temporary entry in his routing table *seqno, Prev RNym*, *Next RNym*, *Prev hop*,*Nexthop*, where *seqno* is the route request sequence number, *Prev Rnym* denotes the route pseudonym of previous hop, *Next RNym* is the route pseudonym of next hop, *Prev hop* is the upstream node and *Next hop* is the downstream node along the route. As any node does not know the real identity of its upstream or downstream node.

entry maintained by *S* temporarily is *(seqno,−,NS,−)*.After that, *S* encrypts these items using its local broadcast key *kS*∗ to obtain *EkS*∗ (*RREQ,NS,ED*(*S,D, rSP*)). Finally, *S* broadcast the following unobservable route request to its neighbors:

NonceS,Nym*S*,*EkS*(*RREQ,NS,ED*(*S,D,rSP*),*seqno*)

Upon receiving the route request message from *S*, *A* tries all his session keys shared with all neighbors to calculate *H*3(*kX*|*NonceS*) or *H*3(*kXA*|*NonceS*) to see which one matches the received Nym*S*. Then *A* would find out *kS*∗ satisfies Nym*S* = *H*3(*kS*|*NonceS*), so he uses *kS*∗ to decrypt the ciphertext. After finding out this is a route request packet, *A* tries to decrypt *ED*(*S,D, rSP*) using his private Idbased key to see whether A is the destination node. To avoid RREQ broadcasting storm, A will check if he has received the same request before by looking up in his cache, which includes a list of *NS* and *seqno*. If it is not a duplicate RREQ, *A* caches *NS* and *seqno* for a given time to detect multiple receipt of the same RREQ packet. In this example, *A* is not the destination and his trial fails, so he acts as an intermediate node. *A* generates a nonce *NonceA* and a new route pseudonym *NA* for this route. He then calculates a pseudonym Nym*A* = *H*3(*kA*∗|*NonceA*). He also records the route pseudonyms and sequence number in his routing table for purpose of routing, and the corresponding table entry he maintained is *(seqno,NS,NA, S, −)*. At the end, *A* prepares and broadcast the following message to all its neighbors:

NonceA,Nym*A*,*EkA*(*RREQ,NA,ED*(*S,D,rSP*),*seqno*)

Other intermediate nodes do the same as *A* does. Finally, the destination node *D* receives the following message from *C*:

NonceC,Nym*C*,*EkC*(*RREQ,NC,ED*(*S,D,rSP*),*seqno*

https://dx.doi.org/10.22161/ijaers.4.2.1 *ISSN: 2349-6495(P) | 2456-1908(O)*

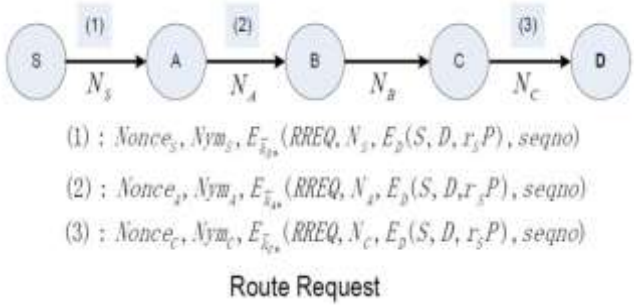

Fig.3: Route Request USOR

Likewise, *D* finds out the correct key *kC* according to the equation Nym*C* = *H*3(*kC*∗|*NonceC*). After decrypting the ciphertext using *kC*∗, *D* records route pseudonyms and the sequence number into his route table. Then *D* successfully decrypts *ED*(*S*,*D*, *rSP*) to find out he is the destination node. *D* may receive more than one route request messages that originate from the same source and have the same destination *D*, but he just replies to the first arrived message and drops the following ones. The route table entry recorded by *D* is *(seqno,NC,−, C, −)*.

Route Reply (RREP): After node *D* finds out he is the destination node, he starts to prepare a reply message to the source node. For route reply messages, unicast instead of broadcast is used to save communication cost. *D* chooses a random number *rD* and computes a ciphertext *ES*(*D, S, rSP, rDP*) showing that he is the valid destination capable of opening the trapdoor information. A session key *kSD* = *H*2(*rSrDP*|*S*|*D*) is computed for data protection. Then he generates a new pairwise pseudonym Nym*CD* = *H*3(*kCD*|*NonceD*) between *C* and him. At the end, using the pairwise session key *kCD*, he computes and sends the following message to *C*:

NonceD,Nym*CD*,*EkCD*(*RREP*,*NC*,*ES*(*D*,*S*,*rSP*, *rDP*),*seqno*)

When *C* receives the above message from *D*, he identifies who the sender of the message is by evaluating the equation Nym*CD* = *H*3(*kCD*|*NonceD*). So he uses the right key *kCD* to decrypts the ciphertext, then he finds out which route this RREP is related to according to the route pseudonym *NC* and *seqno*. *C* then searches his route table and modifies the temporary entry *(seqno,NB,NC,B, −)* into *(seqno,NB,NC,B,D)*. At the end, *C* chooses a new nonce *NonceC*, computes Nym*BC* = *H*3(*kBC*|*NonceC*), and sends the following message to *B*:

NonceC,Nym*BC*,*EkBC*(*RREP*,*NB*,*ES*(*D*,*S*,*rSP*,*rDP*), *seqno*).(5)

Other intermediate nodes perform the same operations as *C* does. Finally, the following route reply is sent back to the source node *S* by *A* in our example illustrated in the Fig. 4:

NonceA,Nym*SA*,*EkSA*(*RREP*,*NS*,*ES*(*D*,*S*,*rSP*, *rDP*), *seqno*). (6)

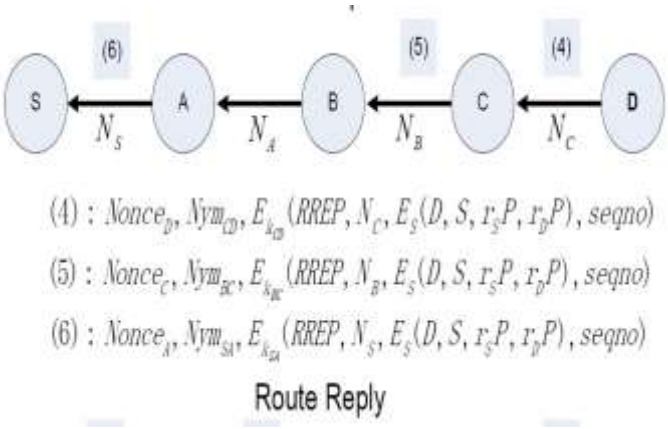

Fig.4: Route Reply USOR

S decrypts the ciphertext using the right key *kSA* and verifies that *ES*(*D, S, rSP, rDP*) is composed faultlessly. Now *S* is ensured that *D* has successfully opened the route request packet, and the route reply is really originated from the destination node *D*. *S* also computes the same session key *kSD* = *H*2(*rSrDP*|*S*|*D*) as *D* does. Till now, *S* has successfully found a route to the destination node *D*, and the route discovery process is finished with success. *S* then finds and modifies his temporary route table entry *(seqno,−,NS,−,−)* into *(seqno,−,NS,−,A)*.

3) Unobservable Data Packet Transmission: After the source node *S* successfully finds out a route to the destination node *D*, *S* can start unobservable data transmission under the protection of pseudonyms and keys. As illustrated in Fig. 5, data packets from *S* must traverse *A*, *B*, and *C* to reach *D*. The data packets sent by *S* take the following format (*DATA* denotes the packet type):

NonceS,Nym*SA*,*EkSA*(*DATA*,*NS*,*seqno*,*EkSD*(*payload*)).(7)

Upon receiving the above message from *S*, *A* knows that this message is for him according to the pseudonym Nym*SA*. After decryption using the right key, *A* knows this message is a data packet and should be forwarded to *B* according to route pseudonym *NS*. Hence he composes and forwards the following packet to *B*:

NonceA,Nym*AB*,*EkAB*(*DATA*,*NA*,*seqno*,*EkSD*(*payload*)). (8)

The data packet is further forwarded by other intermediate nodes until it reaches the destination node *D*. At the end, the following data packet is received by *D*:
NonceC,Nym*CD*,*EkCD*(*DATA*,*NC*,*seqno*,*EkSD*(*payload*)). (9)

https://dx.doi.org/10.22161/ijaers.4.2.1 *ISSN: 2349-6495(P) | 2456-1908(O)*

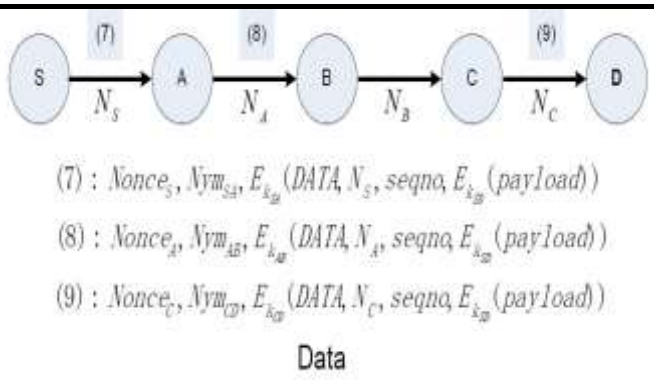

Fig.5: Data Transmission USOR

SHA algorithm and the RSA algorithm is used to encrypt the data. In this routing all the part of the route is maintained secretly like all sequence number, source ID, Destination ID, Packet etc, and this means to provide the good security environment in the routing even in the malicious environment.

III. COMPARISON OF THE PROTOCOLS

Network performance refers to the service quality of a communications product as seen by the customer. There are many different ways to measure the performance of a network, as each network is different in nature and design. The performance of the network mainly refers by using the packet delivery function and the over head of the packet to reach the destination.

1. Packet delivery function: PDF is the term used to measure the network performance. PDF defines the how much amount of packet data delivered to the destination correctly over total number of packets sent by the source. Here we are going to analyze the total number of packets that are delivered to the destination. PDF can be graphed by using the xgraph. First of all calculating the total number of packets deliver to the destination according with the time. So finally we can measure the 10 values and form a graph in the both the routing protocols AODV and USOR. By comparing the two protocols we can analyze the best performance of the protocol.\

2. Overhead: Overhead is the one important concept to analyze network performance. Overhead is defined as number of routing and control packet is requiring transferring the data.

IV. RESULT

In this paper we analyzed the AODV and USOR with the malicious environments with main network parameters such as packet delivery radio and overhead. Result shown below is packet delivery function. In that graph, there are the two environments (AODV with malicious environment and USOR with malicious environment) shown in Figure: 6 and Figure: 7 show the bar chart of overhead.

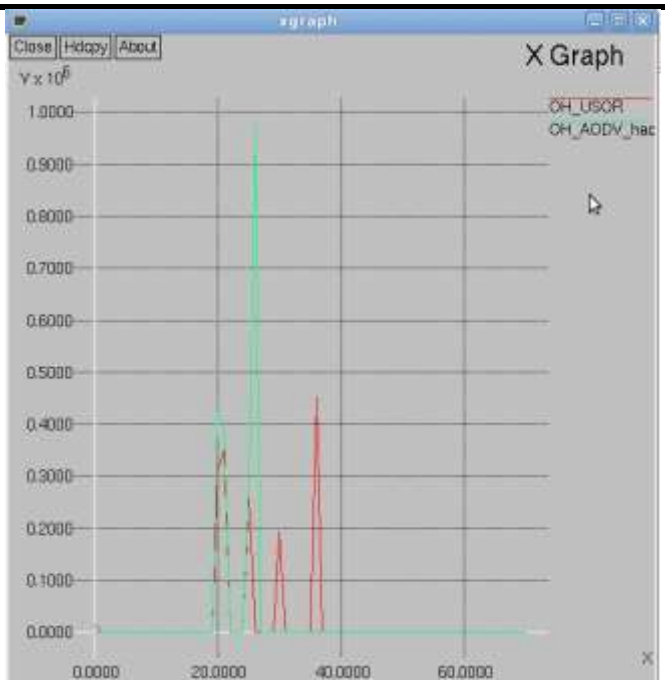

Fig.6: X-graph-overhead comparisons of AODV and USOR

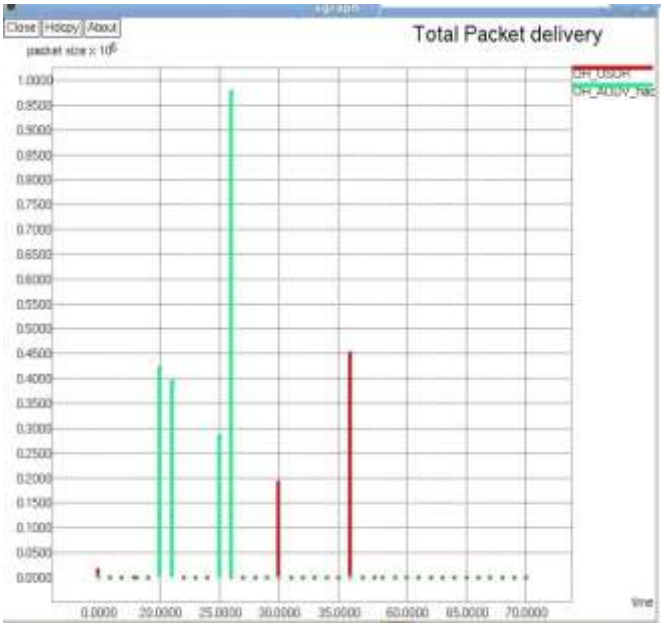

Fig.7: Bar chart-overhead comparisons of AODV and USOR

In the above graph and bar chart shows the overhead of the AODV and the USOR. Blue color lines denoted the AODV and the red color lines denoted the USOR. The overhead of the AODV with the malicious environment is greater than the overhead of USOR with the malicious environment. USOR performance is better than normal AODV even overhead is more; the reason is security of USOR is very high so overhead is ignorable in this case.

The packet delivery function of the both the protocols are shown in figure 7, 8. Here the packet delivery function of the AODV is blue in color and it is give some packet deliver up to some extend and was stop the packet delivery due to the hacking environment. Malicious node could not pass the

https://dx.doi.org/10.22161/ijaers.4.2.1 *ISSN: 2349-6495(P) | 2456-1908(O)*

packet to the destination by the way the packet delivery is minute. In the USOR that is denoted by the red in color, it has more packet delivery at peak level that is the route request and response packets in between the source and destination. Afterwards the packet deliver is in continuous up to the communication ends. So the packet delivery function of the USOR is more effective than in the AODV. By the way USOR provides the secure data communication even in the malicious environment.

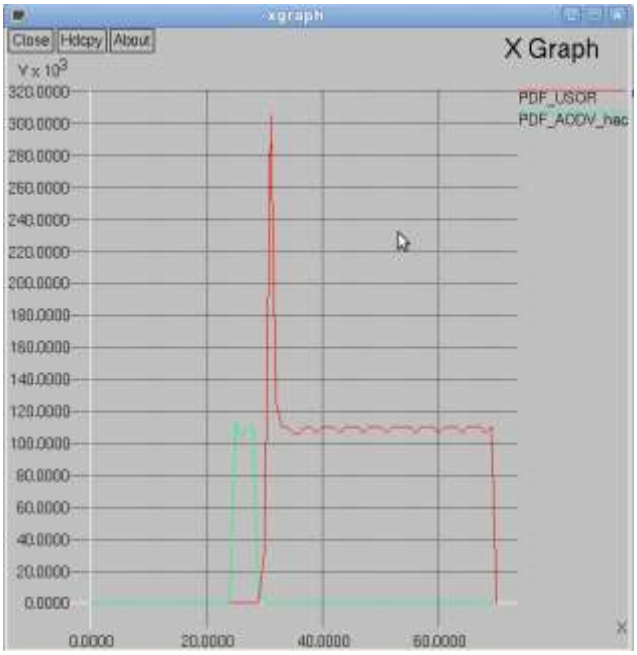

Fig.7: x-Graph-PDF comparisons of AODV and USOR

Fig.8: Bar chart-PDF comparisons of AODV and USOR

USOR provides the strong security requirements to the route and the data that is transfer to the source to destination. Strong security is attained by the unlinkability, unobsevability, secure algorithms RSA, SHA. The unobsevability is kept the all the part of the packet and route information secretly that is source id, destination id, sequence number, packet id etc. this information is kept secret and will known by only the destination when routing process. Secure encryption and decryption algorithms RSA and SHA provide the strong privacy to the information that no one can decrypt the data except destination. Finally the USOR results the strong privacy in routing and information in between the source and destination in mobile ad hoc networks.

V. CONCLUSION

In this paper, we suggested an unobservable routing protocol USOR based on group signature and ID-based cryptosystem for ad hoc networks. The conception of USOR offers solid privacy protection complete unlinkability and content unobservability for ad hoc networks. The protection analysis demonstrates that USOR not only provides strong privacy protection, it is also more resistant against attacks due to node compromise. By the way USOR has satisfactory performance in terms of packet delivery ratio, latency and normalized control bytes than the AODV with malicious environment.

REFERENCES

[1] Conti M, Giordano S. Mobile ad hoc networking: milestones, challenges, and new research directions. IEEE Communications Magazine. 2014 Jan;52(1):85-96.

[2] Belkneni M, Bennani MT, Ahmed SB, Kalakech A. Network Layer Benchmarking: Investigation of AODV Dependability. InInternational Symposium on Computer and Information Sciences 2016 Oct 27 (pp. 225-232). Springer International Publishing.

[3] Vijayan A, Thomas T. Anonymity, unlinkability and unobservability in mobile ad hoc networks. InCommunications and Signal Processing (ICCSP), 2014 International Conference on 2014 Apr 3 (pp. 1880-1884). IEEE.

[4] Belkneni M, Bennani MT, Ahmed SB, Kalakech A. Network Layer Benchmarking: Investigation of AODV Dependability. InInternational Symposium on Computer and Information Sciences 2016 Oct 27 (pp. 225-232). Springer International Publishing.

[5] Sha P, Zhu Z. The modification of RSA algorithm to adapt fully homomorphic encryption algorithm in cloud computing. InCloud Computing and Intelligence Systems (CCIS), 2016 4th International Conference on 2016 Aug 17 (pp. 388-392). IEEE.

[6] Aarthi G, Ramaraj E. Hybrid Encryption Technique Using RSA with SHA-1 Algorithm in Data-At-Rest and Data-in-Motion Level. International Journal of Computer Science and Information Security. 2014 Jun 1;12(6):55.

[7] KM MK, Sunitha NR, Mathew R, Veerayya M, Vijendra C. Secure Ad-Hoc On-demand Distance Vector routing using identity based symmetric key

https://dx.doi.org/10.22161/ijaers.4.2.1 *ISSN: 2349-6495(P) | 2456-1908(O)*

management. InWireless Communications, Signal Processing and Networking (WiSPNET), International Conference on 2016 Mar 23 (pp. 1075-1081). IEEE.

[8] Narasimhan B, Balakrishnan R. Energy Efficient Ad-Hoc On-Demand Distance Vector (Ee-Aodv) Routing Protocol For Mobile Ad Hoc Networks. International Journal of Advanced Research in Computer Science. 2013 Jul 1;4(9).

[9] Balaji S, Rajaram M. SIPTAN: Securing Inimitable and Plundering Track for Ad Hoc Network. Wireless Personal Communications. 2016 Sep 1;90(2):679-99.

[10] Liu W, Yu M. AASR: authenticated anonymous secure routing for MANETs in adversarial environments. IEEE transactions on vehicular technology. 2014 Nov;63(9):4585-93.